65



Documentation On

mediQuick

PG-DAC Sept 2023

# Group No: 16

|  |  |
| --- | --- |
| **Roll No** | **Name** |
| **243032** | **Gayatri Naval Kungade** |
| **243072** | **Rahul Singh** |

\

# Mrs. Vaishnavi Ghodke Mr. Rohit Puranik

**ABSTRACT**

This project focuses on the development of a web-based shopping system tailored for an existing medical store, aiming to bridge the gap between traditional brick-and-mortar pharmacy shopping and modern e-commerce convenience. The objective is to provide customers with the flexibility and accessibility of online shopping while maintaining the reliability and quality assurance associated with a trusted medical establishment.

The web platform enables customers to browse an extensive catalog of medicines, select desired items, and conveniently place orders from any internet-enabled device. With features such as user registration, product search and filtering, secure online transactions, and order tracking, the system ensures a seamless and user-friendly shopping experience.

By integrating home delivery services and offering a wider selection of medicines, the project strives to enhance customer satisfaction and convenience. Additionally, provisions for customer support channels and feedback mechanisms aim to foster trust and engagement with the online platform.

**ACKNOWLEDGEMENT**

I take this occasion to thank God, almighty for blessing us with his grace and taking our endeavor to a successful culmination. I extend my sincere and heartfelt thanks to our esteemed guide, **Mrs. Vaishnavi Ghodke** for providing me with the right guidance and advice at the crucial juncture sand for showing me the right way. I extend my sincere thanks to our respected Centre Co-Ordinator **Mr. Rohit Puranik**, for allowing us to use the facilities available. I would like to thank the other faculty members also, at this occasion. Last but not the least, I would like to thank my friends and family for the support and encouragement they have given me during the course of our work.

Gayatri Naval Kungade (240341220064)

Rahul Singh (240341220148)

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Title** | **Pg. No.** |
| 1 | INTRODUCTION | 1 |
| 1.1 | PURPOSE | 2 |
| 1.2 | SCOPE | 2 |
| 1.3 | OBJECTIVE OF PROJECT | 2 |
| 1.4 | FUNCTIONALITIES | 3 |
| 2 | REQUIREMENT | 4-6 |
| 3 | DATABASE DESIGN | 7-11 |
| 4 | APPENDIX A | 12-15 |
| 4.1 | ER Diagram | 12 |
| 4.2 | Class Diagram | 13 |
| 4.3 | Data Flow Diagram | 14 |
| 4.4 | Activity Diagram | 15 |
| 4.5 | Use Case Diagram | 19 |
| 5 | APPENDIX B | 16-23 |
| 6 | References | 24 |

# INTRODUCTION

Introducing mediQuick.com, your premier destination for convenient and secure online pharmaceutical shopping. Designed to streamline the process of purchasing essential healthcare products, mediQuick.com offers an interactive platform where users can seamlessly navigate through a diverse array of medical supplies. From prescription medications to over-the-counter remedies, our website provides a user- friendly interface that ensures a hassle-free shopping experience. With a focus on security, efficiency, and expanded product categories, we aim to redefine the way customers access healthcare products online. Join us on mediQuick.com and experience the future of online pharmacy shopping today.

### Purpose

The aim of this document is to explain the functionality of the project developed for Online medicines ordering using UI (mediQuick) This document will provide a baseline for design of the database, user interfaces, coding & evaluation of test plans. It will be used as a solid foundation for continued product evaluation.

Technologies we have used here is Spring-Boot, React-Js and MySQL.

### Scope

Online Medicine Ordering System will be always available for customers where they can explore the variety of products. The system provides secure registration and profile management facilities for customers. It will be having adequate searching mechanisms for easy and quick access to particular products. The system also provides the easy solution for the registered customers to buy the product online using internet and also allow them to maintain their cart for adding or removing the product.

### Objectives:

The objective of this document is to deliver a detailed depiction of the Online Medicine Ordering System. It will explain the function and characteristics of the system, the boundaries and purpose of system, and all external environment restrictions under which this system must operate and react successfully.

The software System, an Online Medicine Ordering System(OMOS) will be designed for an ordering purpose. The purpose of this system is to combine all the database of medicines as well as the records of sales, purchases. It will also help user to search for medicines, price, and their availability.

More specifically, this designed system will allow registered user to search for specific drugs and order online. Online payment option along with cash on delivery is available to users.

### Functionalities provided by mediQuick

mediQuick is an online based e-commerce website, which is an outstanding way of bringing customers on a online platform to make purchase in a secured and efficient manner. This website provides an Interactive interface through which a user can interact with different areas of the application easily by maintaining the product’s as well as customer’s information. OMOS provides a simple interface and platform to ease the process of buying the products online. It includes smooth functionality and efficiency that adds to buyers’ confidence. Medicart.com keeps a constant focus on new category creation and expansion of products.

Users are classified into four parts, the Administrator and Customer, Supplier and Delivery-Boy

* **Customer:** He or she is a verified user of website who is intended to buy a product via OSS. The customer must have a username and password to make a purchase. If the customer is not registered, they should register themselves by navigating to the sign up page.
* **Administrator:** He or she is a verified person who is allowed to sell items over the platform. Admin is responsible for monitoring functions and procedures on platform.
* **Supplier :** The group of people or a single person which is registered with platform is responsible for supplying the product in bulk quantity on regular basis or as required, as a supplier he or she would be able to see orders and dispatch the batch of products.
* **Deliver-Boy** : He or she is a verified employee or 3rd party partner who will be key member of business model to deliver the products at the customer end. As a Delivery boy he/she will be able to see Assigned orders.

# REQUIREMENTS

### Functional Requirements

1. User Registration and Authentication:
   * Users can register by providing necessary details.
   * User authentication is required to access the system.
2. Product Management:
   * Admins should be able to add new products to the system with details like name, description, price, and image.
   * Admins should have the ability to delete, update, and view existing products
3. Order Management:
   * Users should be able to add products to their cart and proceed to checkout.
   * Admins should have access to order details, including order status, payment information, and shipping details.
   * Admins should be able to process orders, update order statuses, and manage order fulfilment.
4. User Management:
   * Admins should have the ability to manage user accounts, including creating, updating, and deleting accounts.
   * Users should be able to update their profile information, including shipping addresses and contact details.
   * Admins should be able to view and manage user feedback, reviews, and complaints.

### Non-Functional Requirements:

1. Security:
   * User passwords are securely stored using encryption techniques.
   * Access controls ensure that users can only access authorized features.
2. Performance:
   * The system should handle a large number of simultaneous users without significant slowdowns.
   * Image loading and retrieval should be efficient for a smooth user experience.
3. Scalability:
   * The system should be designed to accommodate future growth and increased user activity.
4. Usability:
   * The user interface should be intuitive and user-friendly for both customers and administrators.
   * Clear and concise error messages should guide users through any issues.
5. Reliability:
   * The system should be available and operational 24/7 with minimal downtime.
6. Data Integrity:
   * Data integrity and consistency are maintained through proper validation and database design.
7. Data Privacy:
   * User data, especially personal and sensitive information, should be stored securely.

### Other Requirements:

Hardware and Network Interfaces: Back-end Server Configuration:

* + Intel Pentium-IV Processor
  + 128 MB RAM
  + 1 Raid Controller Card
  + 32-bit Ethernet Controller (100 Base-T)
  + 8 x 2.0 GB Fast SCSI/2 with Raid Support
  + 2.88 MB FDD
  + 48x CD ROM Drive
  + SVGA Colour Monitor on PCI with 1MB RAM
  + 101 Keys Keyboard
  + 1 Microsoft Mouse with pad
  + 4/8 GB DAT
  + One Serial & Two Parallel Ports
  + Internet Information Server (IIS)
  + Microsoft Transaction Server (MTS)

Front-end Client Configuration:

* + Intel Pentium-III @ 650 MHz Processor
  + 128 MB SDRAM
  + 10 GB Hard Disk Drive
  + 1.44 MB Floppy Disk Drive
  + 15” SVGA Digital Color Monitor
  + One Serial, One Parallel port, and One USB port
  + 104 Keys Keyboard
  + PS2 Mouse with pad
  + 32-bit PCI Ethernet Card
  + 48X CD Drive Software Interfaces:

Software configuration for back-end Services:

* + Windows NT – Server 4.0
  + SQL Server 8.0

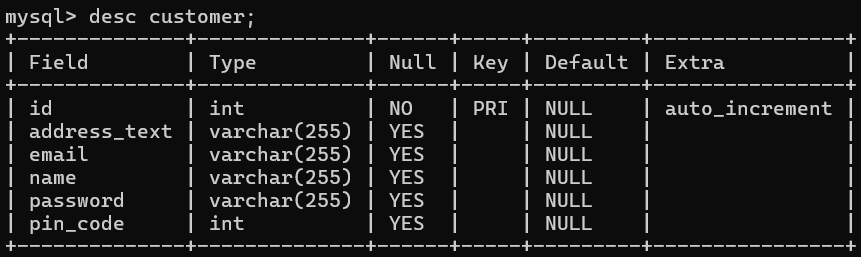
- STS – 3.9.18

Software configuration for front-end Services:

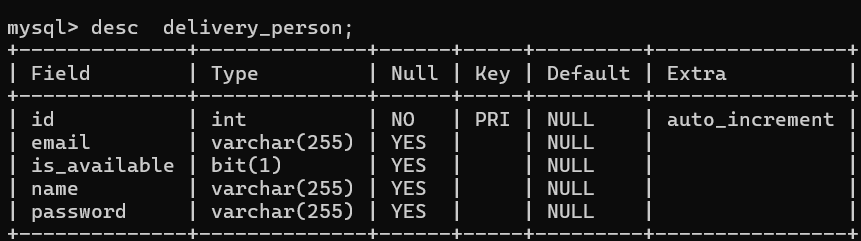
* + Virus Protection Software
  + Client Workstation
  + Office 2000 Web Browser:
  + Internet Explorer, Crome

## DATABASE DESIGN

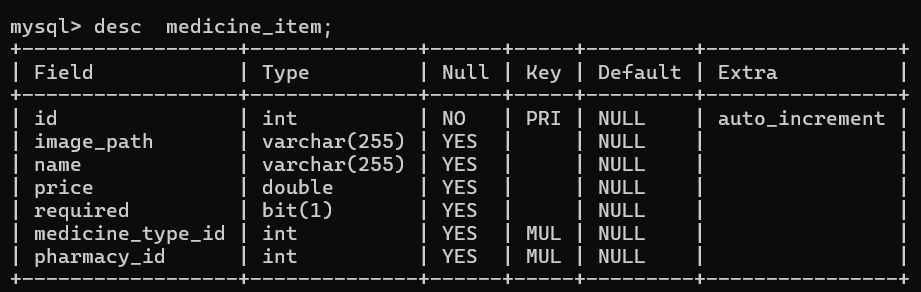
* Customer:



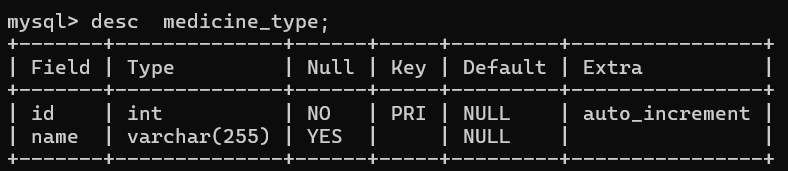
* delivery-person:



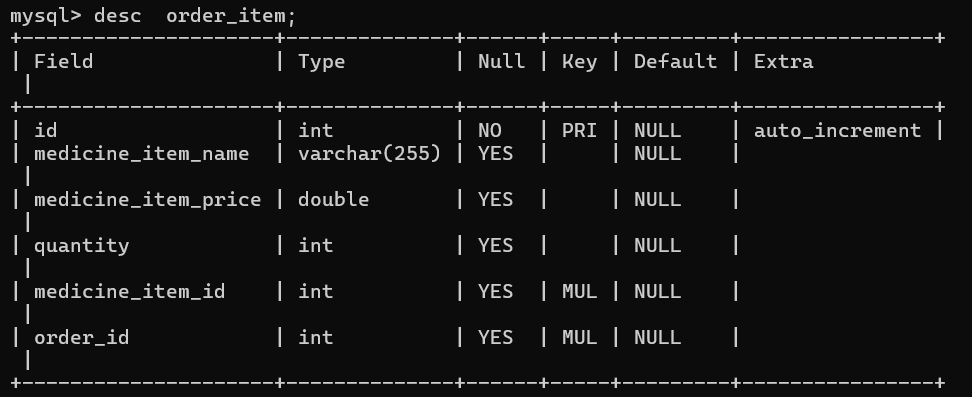
* medicine\_item



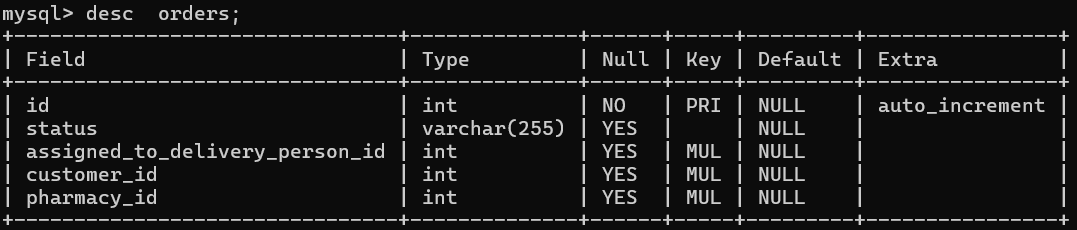
* medicine\_type



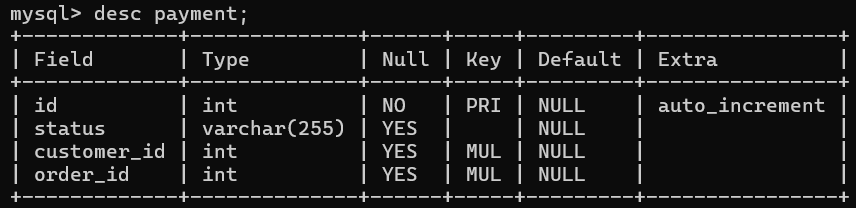
* order\_item



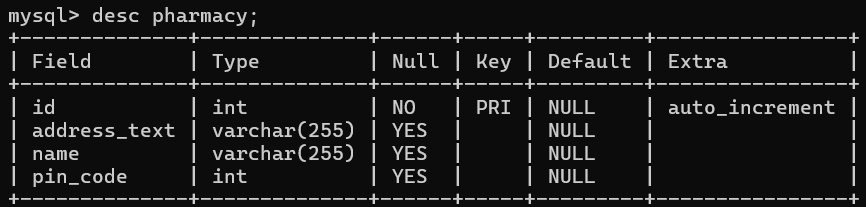
* orders



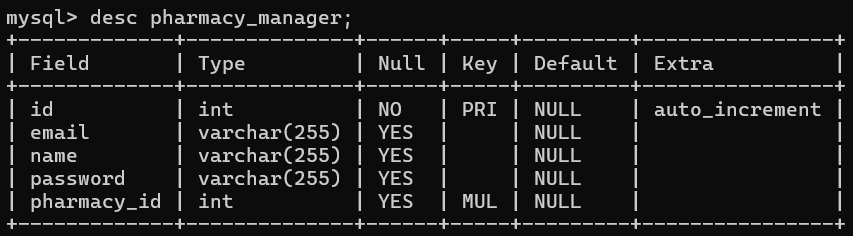
* payment



* pharmacy



* pharmacy\_manager



## DATABASE

Databases are the storehouses of data used in the software systems. The data is stored in tables inside the database. Several tables are created for the manipulation of the data for the system. Two essential settings for a database are

* Primary key - the field that is unique for all the record occurrences
* Foreign key - the field used to set relation between tables Normalization is a technique to avoid redundancy in the tables

Normalization is a technique to avoid redundancy in the tables.

## SYSTEM TOOLS

The various system tools that have been used in developing both the front end and the backend of the project are being discussed in this chapter.

## FRONT END:

React is a library which is developed by Facebook are utilized to implement the frontend. React (also known as React.js or ReactJS) is a free and open-source front-end JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.

## BACKEND:

The back end is implemented using MySQL which is used to design databases.

### MySQL:

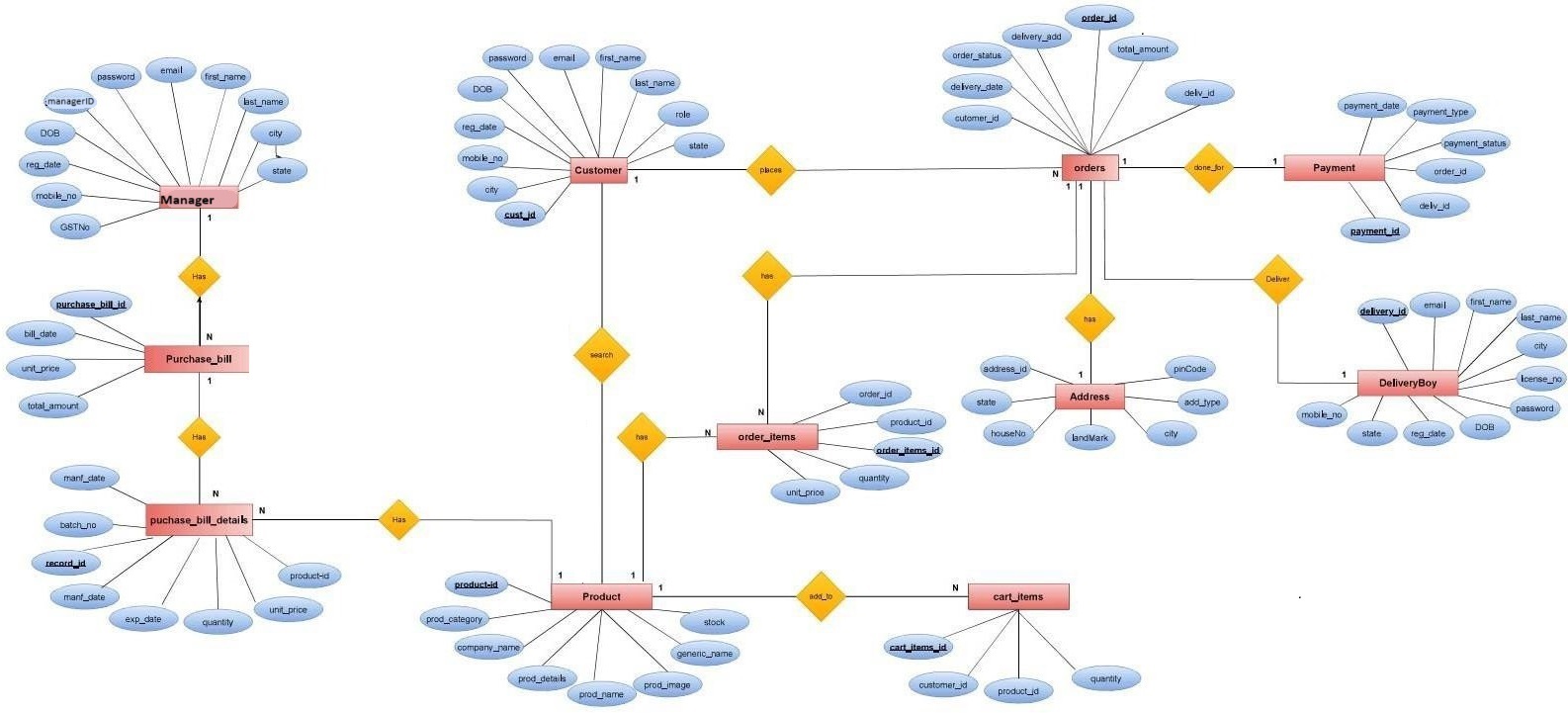
MySQL is the world's second most widely used open-source relational database management system (RDBMS). The SQL phrase stands for Structured Query Language. An application software called Navicert was used to design the tables in MySQL.

### Spring-Boot:

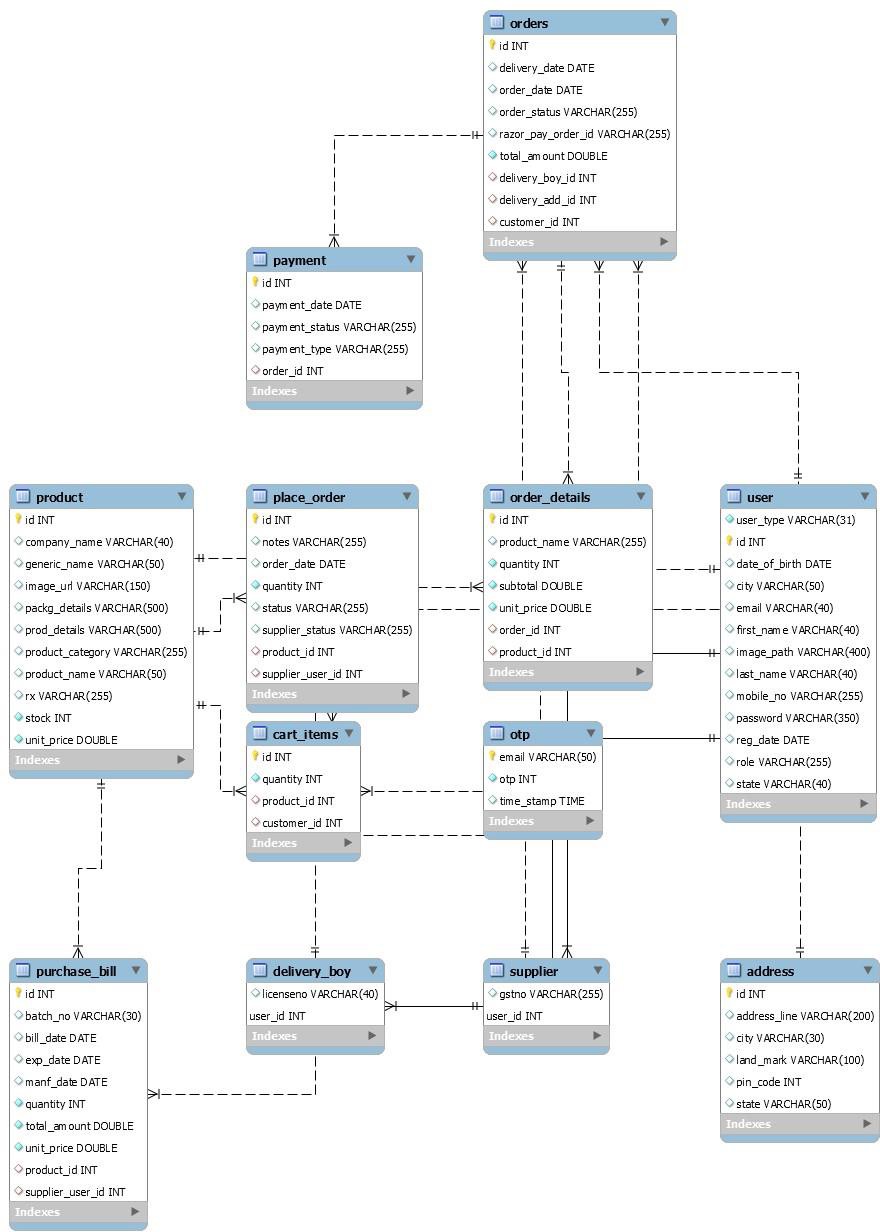
This is used to connect MYSQL and fetch data from database and store the data in database. The Spring Framework is an application framework and inversion of control container for the Java platform. The framework's core features can be used by any Java application, but there are extensions for building web applications on top of the Java EE (Enterprise Edition) platform. Although the framework does not impose any specific programming model, it has become popular in the Java community as an addition to the Enterprise JavaBeans (EJB) model. The Spring Framework is Open- source Framework.

## APPENDIX A

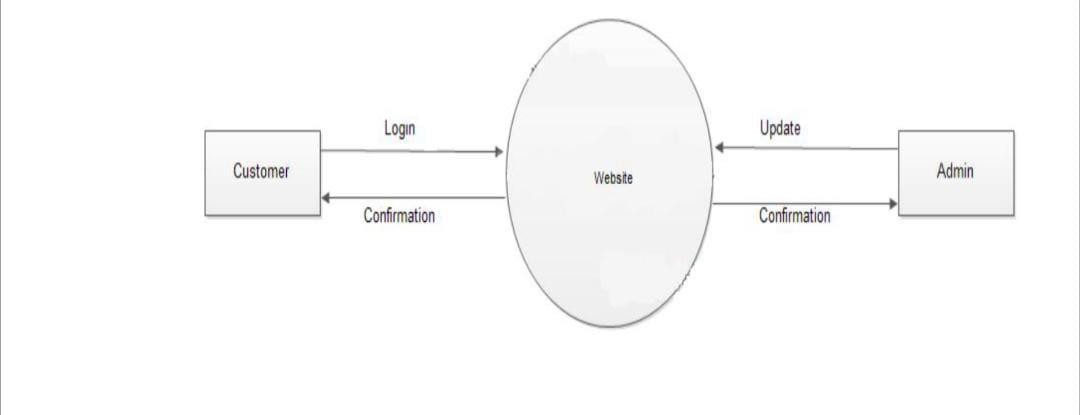
Entity Relationship Diagram:



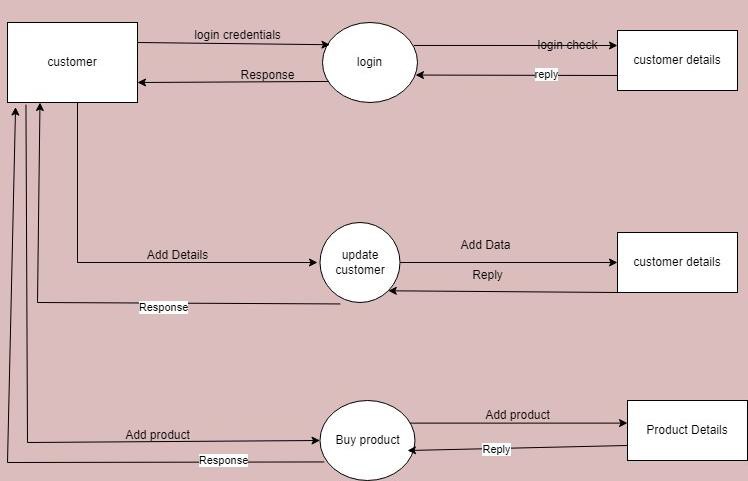
Class Diagram



Data Flow Diagram:

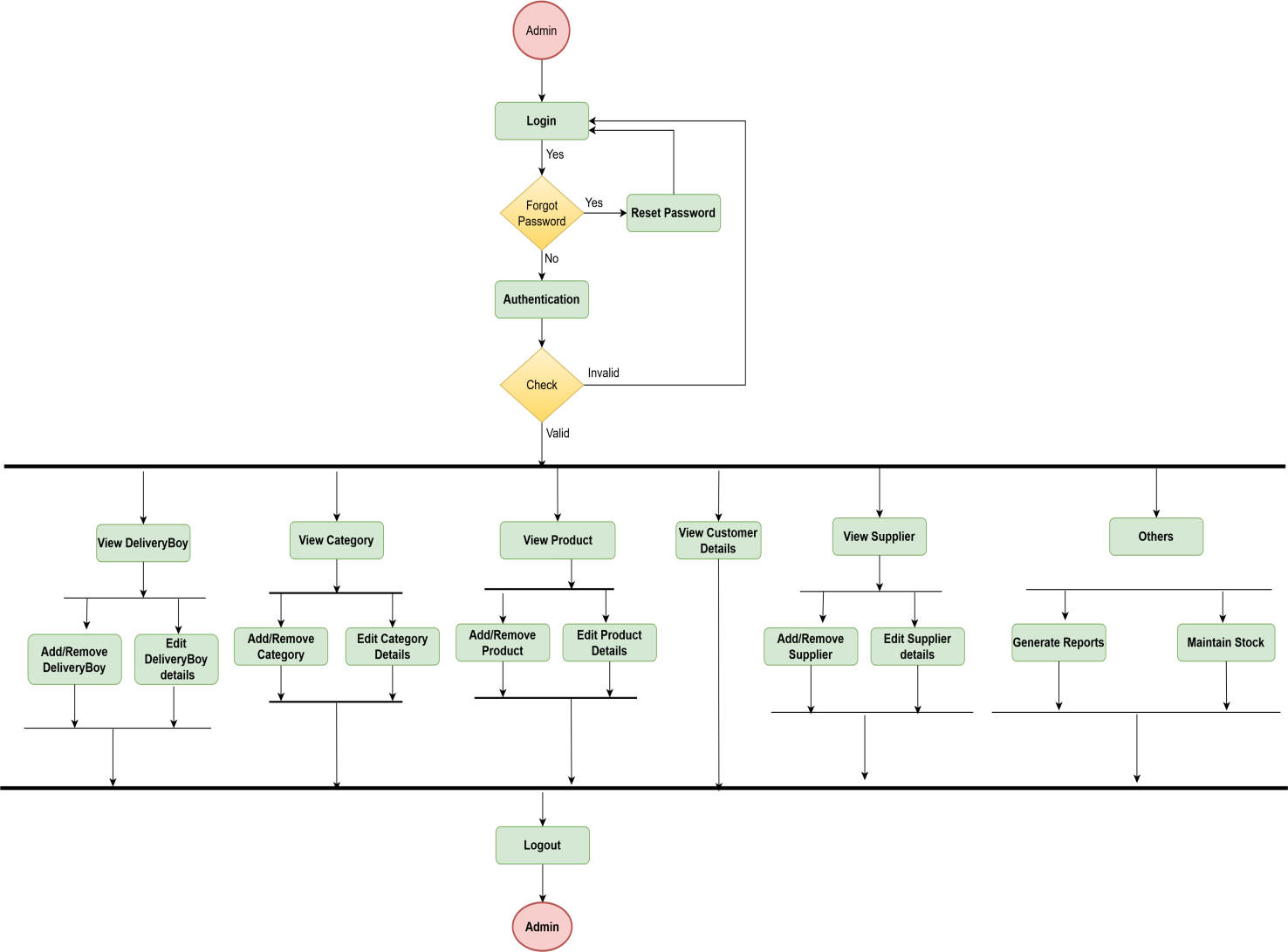
Level 0:

Level 1:

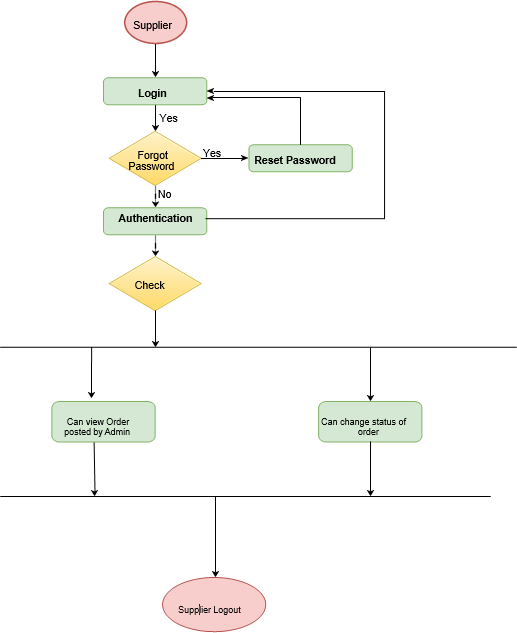


Activity Diagram

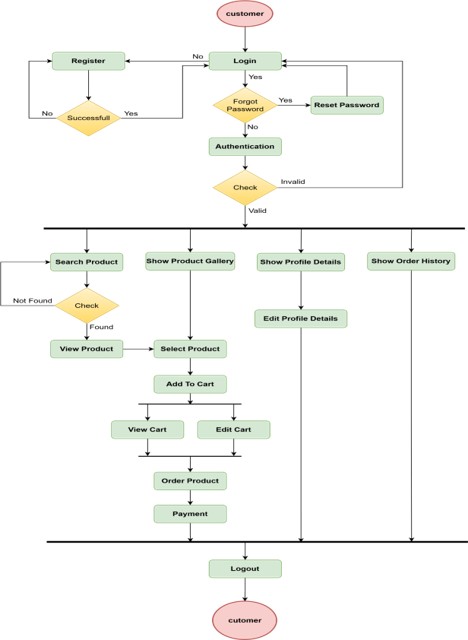
1. Admin



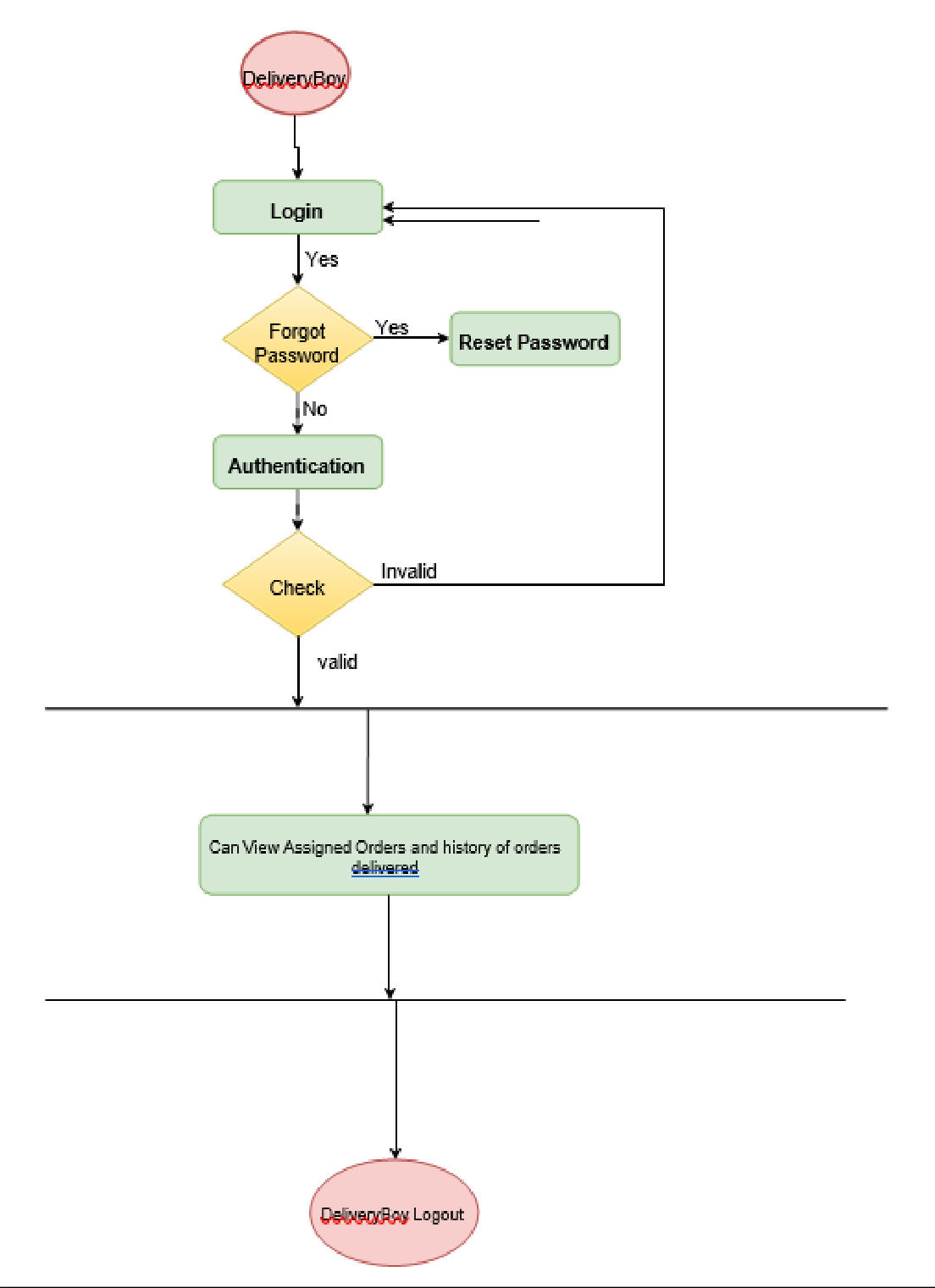
1. Supplier



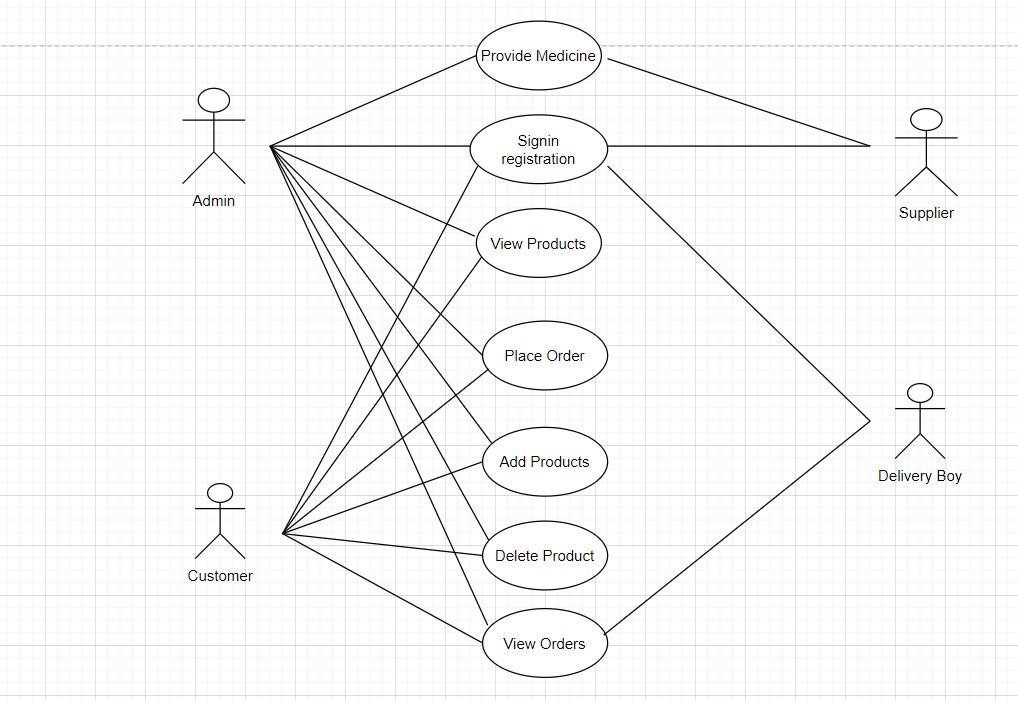
1. Customer



1. Delivery Boy



Use Case Diagram:



1. APPENDIX B

### References

<http://www.javatpoint.com/javatutorial>

<https://www.tutorialspoint.com/java>

<https://www.baeldung.com/learn-jpa-hibernate>